

# CLOSED LOOP TUNING VS OPEN LOOP TUNING: TUNING ALL YOUR LOOPS WHILE THE PROCESS IS RUNNING IS NOW POSSIBLE

**Michel Ruel**

Top Control Inc., 49, Bel-Air, #103  
Levis, Qc, Canada G6V 6K9  
[mruel@topcontrol.com](mailto:mruel@topcontrol.com)

**Keywords:** Process Control, Optimization, Oscillation, Model, PID Control, Closed Loop, Open Loop, Stiction, Backlash, Non linearity, Pseudo-random-binary Sequence, Generalized Binary Noise

## ABSTRACT

Traditional methodology for optimizing and tuning PID loops (excluding ‘trial-and-error’) rely on ‘open-loop’ tests, whereby the loop is placed in manual mode and the controller output is moved, usually in a step-wise fashion. The issues with this ‘step’ test are numerous, primarily that the test will disturb the process. This is especially true for slow processes such as temperature, where a seemingly small output move could result in a large process change that can have serious consequences.

There are now tools and methods that enable one to safely optimize and tune PID loops in closed loop mode. Some of these tools can generate and send small, rapid setpoint changes to the controller, independently of any operator intervention. With the right understanding of these tools, it is now possible to optimally tune every PID loop in a plant with minimal time and risk. If planned correctly, process control personnel will define the boundaries and conditions and the tests will be done during night shifts or at any moment. Later, the process control engineer will analyze the results and decide on tuning objectives; software will calculate tuning parameters. Optimizing processes and tuning loops without spending hours in the control room reduces not only the resources needed but also production losses and the attention of operators.

This paper will discuss how these new tools may work in closed loop tuning mode, when and how they can be used, when they may fail, and the mistakes people may make when using them.

## INTRODUCTION

Open-loop tests disturb the process and require the attention of operators. Other open-loop tests can be used, such as so-called pulse and double-pulse tests, in conjunction with software tools.

A double pulse test, when executed properly, can help move a process variable back toward setpoint and shorten the test time. A single pulse test does not offer this advantage, but shares the benefit of directing the process variable back to the original value before it deviates too far from normal conditions.

The issues with ‘pulse’ tests are fewer than with ‘step’ tests but still exist. These tests will disturb the process. The loop

must also be closely monitored while in manual mode. The complexity of the double pulse test requires additional skills. Because of these problems and the lack of skills, many people resort to ‘trial-and-error’, whereby the tuning values are changed, and the response to a setpoint step observed. This process is repeated again and again until the response is satisfactory. Should one think clearly about this, they would agree that repeated setpoint step changes while testing tuning values are potentially more disruptive than a few output pulse changes made in manual mode.

## METHODOLOGY OF CLOSED LOOP TUNING

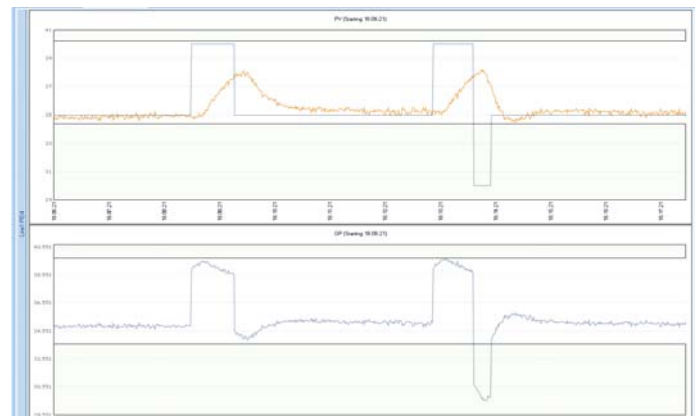
Any tuning method seeks to establish a “cause and effect” between the controller output and the process variable. To do this in open loop, the output is moved directly. In closed loop,

making a change in the setpoint causes the output to move indirectly. One key point for either method is that the response of the process variable must be due solely to the movement of the controller output whether due to a setpoint change in closed loop or a controller output change in open loop. If the process variable movement is due to a disturbance, then the test and the data are invalid for determining any process models and tuning. The exception to this is if you are able to measure and factor in the disturbance(s) or make a setpoint change that causes a process movement larger than that caused by the disturbance. If disturbances occur occasionally, using more data (so that disturbances represent a small portion of the data) will permit the process to be modeled even with occasional disturbances.

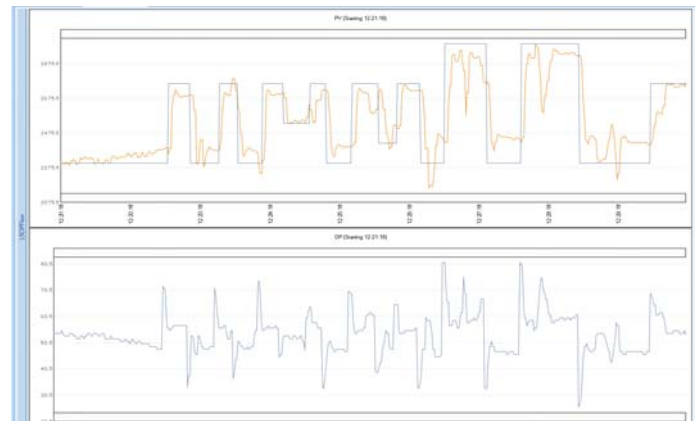
Several types of tests are typically used for closed loop tuning. Changing the setpoint up or down in a step-wise fashion is one such test. This test is likely to be as disruptive as an open loop test unless the setpoint change is part of the normal operation of the process. The only advantage over an open loop test is that unless the loop is tuned to be unstable, it will eventually settle at the new setpoint, a value that you have selected as a safe eventual target. Changing the setpoint in a pulse, or double pulse test, is another type of test. These tests have an advantage over the step test since you return the setpoint to its normal value. The double pulse test has a further advantage in that, when done correctly, the process variable is forced back toward setpoint more rapidly than with a single pulse. All loops where enough setpoint changes are present (amplitude above noise level) can be tuned using historical data if the period of data collection is fast enough.

A pseudo-random-binary sequence (PRBS) is a type of test that is similar to doing multiple double pulse tests. Like with the double pulse test, the process is maintained at the same average as the setpoint, thus eliminating any deviations from normal operation. PRBS is richer and will generate better models than single or double pulse tests.

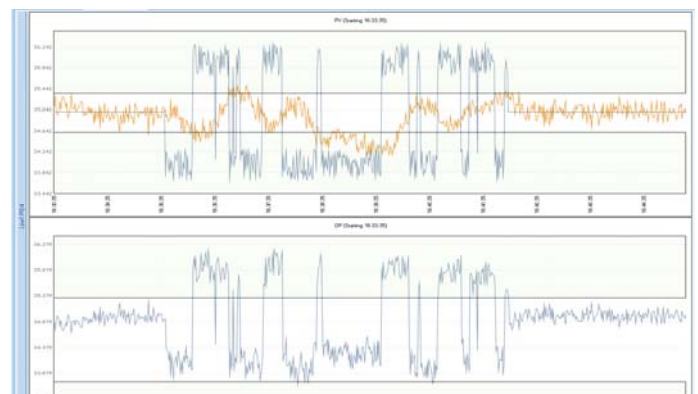
Another alternative is the generalized binary noise (GBN) approach, which is more or less a low frequency version of a pseudo random binary sequence (PRBS) test. A high frequency dither is added on top of the low-mid frequency pattern to improve dead time estimation (which is a high frequency component).



**Figure 1:** Example of closed loop pulse and double pulse tests. The setpoint and process variable are shown in the top plot and the controller output is shown on the bottom.

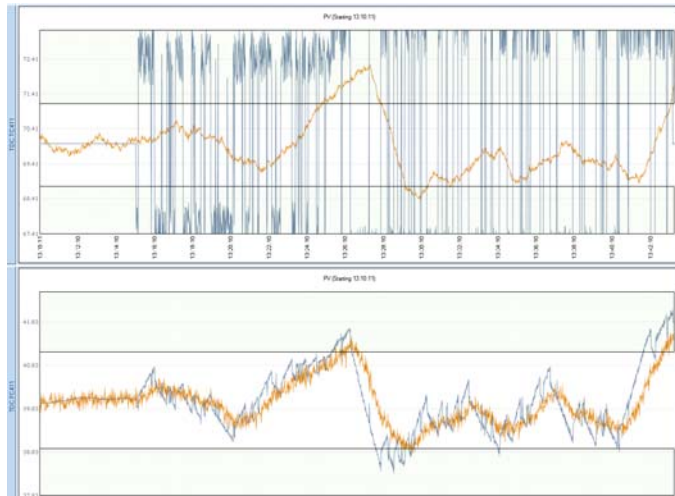


**Figure 2:** Example of a PRBS-like test generated by having the operator move the setpoint. In this data there are many rapid small disturbances from multiple sources (flow path switching) in a clean-in-place (CIP) circuit.



**Figure 3:** Example of an automatically generated closed loop generalized binary noise test with a high frequency dither. The setpoint and process variable are shown in the top plot and the controller output is shown on the bottom.

If setpoint changes occur in a process as a normal part of the process operation, this data can be used to modelize the process and tune the loop. Setpoint changes could originate from operators, APC (advanced process control), other controllers (cascade strategies, ratio loops...). For example, in a cascade control strategy, moving the setpoint on the master will generate enough changes in both loops (cascade and slave) to tune them.



**Figure 4.** Example of an automatically generated closed loop generalized binary noise test with a high frequency dither on a cascade system. The master temperature loop (setpoint and process variable) are shown in the top plot and the secondary flow output is shown on the bottom.

When analyzing cascade loops, using the right software will handle secondary and master tuning parameters; if the user changes values or objectives in the secondary loop, tuning parameters for the master will be automatically recalculated.

**OPEN LOOP OR CLOSED LOOP**

Traditionally, loops were tuned in open loop using bump tests. Most tuning tools use this technique. When analyzing process data in open loop, one must be careful since everything that is programmed in the control system could be missed. For example, controller execution time, filters, ramp limiters, and characterizers will not be included in the test.

Also, simple bump tests require special techniques to detect process defects such as dead band, backlash, hysteresis, stiction, and non linearities.

When analyzing in closed loop, controller sampling time is included, process defects are part of modeling (if setpoint

moves in both direction), and special configuration and programming are included.

Hence, when using closed loop testing techniques, results obtained include every part of the loop and if the tool is powerful enough, even non linearities will be included in the model. Non linearities are usually amplitude dependent. Also, the amplitude of setpoint changes should reflect usual process variable excursions from setpoint.

**DATA SUITABLE FOR CLOSED LOOP TESTING**

What is needed to have sufficient data to modelize a process? The first simple answer is: moves in the controller output sufficient to cause process variable movement superior to noise.

Test	Operations	Process Control	Results
Bumps in manual mode	Operations is involved and needs to supervise tests	Work with operations. Decide on tests ...	Vary with skills and tools; specialists must choose adequate series of bumps or pulses to detect process problems
Bumps on setpoint	Operators understand it and are involved	Requires process control attention and coaching	
Pulses on setpoint	Simple but operators need coaching		Disturb process and operation
Setpoint changes, normal operation		Verify if setpoint changes are sufficient	Vary with skills and tools
PRBS on setpoint	No intervention from operations	Set-up and configuration initially	Modeling, tuning, non linearities, evaluation are automated
GBN with dither			

## BENEFITS OF CLOSED LOOP TUNING

There are two key benefits to closed-loop PID testing. First, the process variable can be kept close to normal operating conditions and therefore cause less disruption to the process. Second, since the loop is in automatic mode, there is no need for dedicated personnel to spend time watching the loop closely, as would be the case in manual. In manual mode, a step test could result in the process variable moving too far from a desired value. A disturbance could have the same affect while the PID is open-loop. In automatic, a safe target can be set, and the loop has some chance at recovery from a disturbance.

If software is used that can generate a test signal for the setpoint movement, even less effort is required by engineering and operating personnel. For the software tests mentioned above, such as PRBS-like and GBN-type tests, the amplitude, width, and duration of the setpoint changes can be safely specified to be within desirable operating conditions. If limits are reached, tests can be stopped automatically or setpoint changes can be limited to remain within limits. If operators move the setpoint during a test, GBN will continue to be added to the new value.

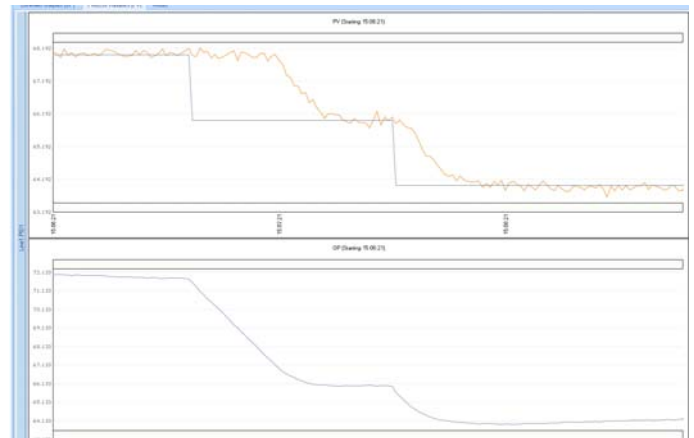
Multiple loops can be tested and tuned at the same time. It becomes doable to tune all loops in a plant.

## WHEN CLOSED LOOP TUNING MAY FAIL

When using closed loop tuning methods, several conditions could lead to unsuccessful tests. For example, controller output hitting minimum or maximum, process variable moving too far away from setpoint, disturbances occurring during a test, special conditions occurring (pump stops, flow paths changing, etc.). Good tools may handle those cases and exceptions. If the loop is in manual and has never been placed in auto or is unstable in auto due to the current tuning or other problems, or the process variable is moving because of a disturbance, with a steady setpoint, then closed loop tuning cannot be used. In the latter case, even open loop methods would not work.

When using closed loop testing, the amplitude of changes used should represent situations similar to usual disturbances and changes. Indeed, if non linearities are present, tuning and modeling tools should handle those non linearities. Valve problems such as hysteresis and stiction, for example, will increase dead time identified and result in more sluggish tuning parameters. Non linearities such as non linear gain or

varying dead time will modify the real model and good tools should suggest more sluggish tuning to guarantee stability.



**Figure 5:** Two setpoint changes for a flow loop are shown in this figure. Note that the process dead time appears to be larger in the first setpoint change than in the second setpoint change. This is due to hysteresis in the valve.

This example will be modeled by software and an error bound will be calculated to include non linearities.

## TOOLS AVAILABLE FOR CLOSED LOOP TUNING

Despite the hype made by salesmen for tools that “automatically” tune loops, the truth is that these tools often do not work as advertised. Some skills are still required to understand when these tools may and may not work; this is one element of this paper.

Next generation tools will handle interacting loops all at once, and modelize the interaction between these loops.

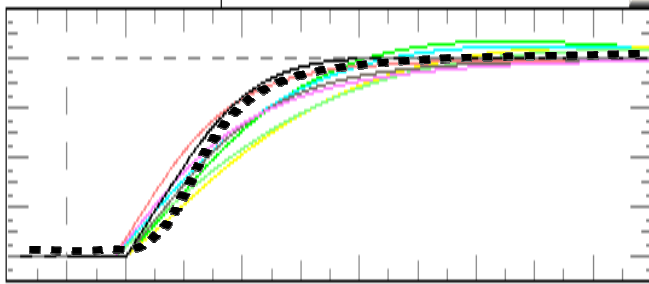
When analyzing a control loop, it is a good practice to check if the model varies with: amplitude, direction, range (e.g. low flow and high flow), and process conditions. To detect those non linearities, tests must include different amplitudes, directions, range.

Most tools will then ask the user to analyze separately each test. After finding the worst case, the user will use this worst case scenario to optimize the process and tune the loop.

With new tools, all tests are analyzed using one stream of data and the model is found including an error bound to take in account those non linearities.

## HANDLING NON LINEARITIES WITH MODERN TOOLS

If the model of a control loop varies, which model should be used for tuning? Non linearities and model changes are taken into account in new modern tools. Figure 6 illustrates different response for the same control loop.



**Figure 6:** Different responses for the same setpoint change.

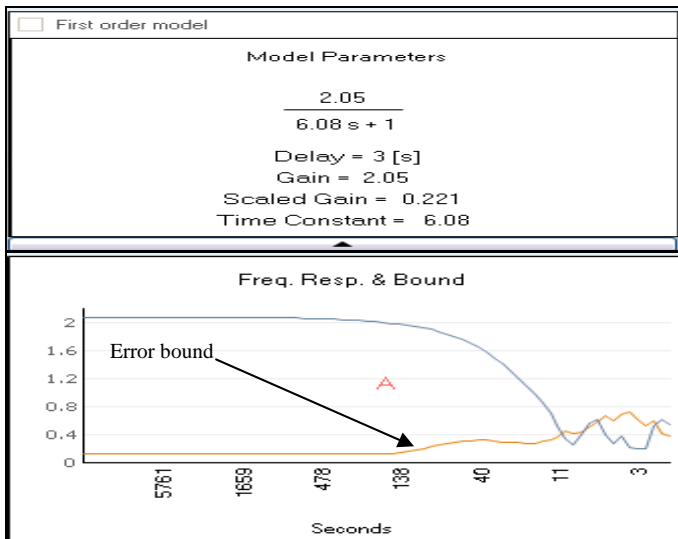
The worst case is (dotted line):

- Maximum dead time
- Maximum speed in closed loop
- Maximum overshoot (process gain)

The worst case model would be:

- Maximum dead time
- Minimum time constant
- Maximum process gain

Software finds a model and uncertainty (non linearities and model variations). This uncertainty or error bound is displayed on a Bode plot.

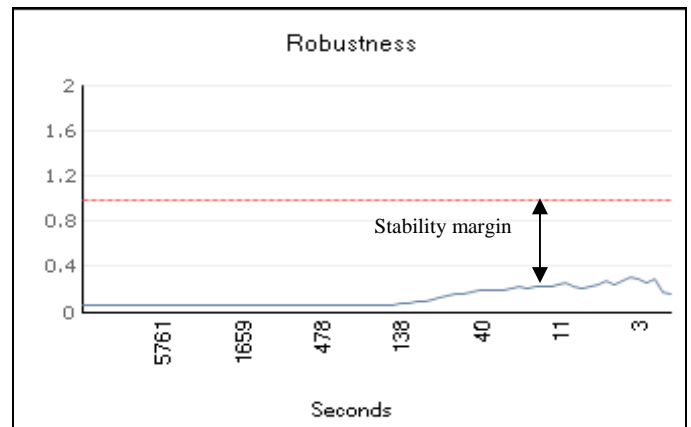


**Figure 7:** Model found and error estimated at all frequencies.

On figure 7, we observe that uncertainty is small at low frequency (long period) but is higher at high frequency (short period), usually corresponding to varying dead time.

When calculating tuning parameters, a robustness plot is displayed; this robustness plot illustrates how robust the tuning parameters are. Robustness is defined as tolerance to instability. It depends on the tuning parameters, the model and the error bound (uncertainties).

The distance between the graph and the horizontal line is the margin to instability at each frequency. For example, in figure 8, the error bound is small at low frequency (long period), then it increases at around 138 seconds. Finally, at high frequency (short period), it is higher. When calculating tuning parameters, if error bound is present, then tuning parameters have to be less aggressive to guarantee stability.



**Figure 8:** Robustness calculated using tuning parameters, model and uncertainties.

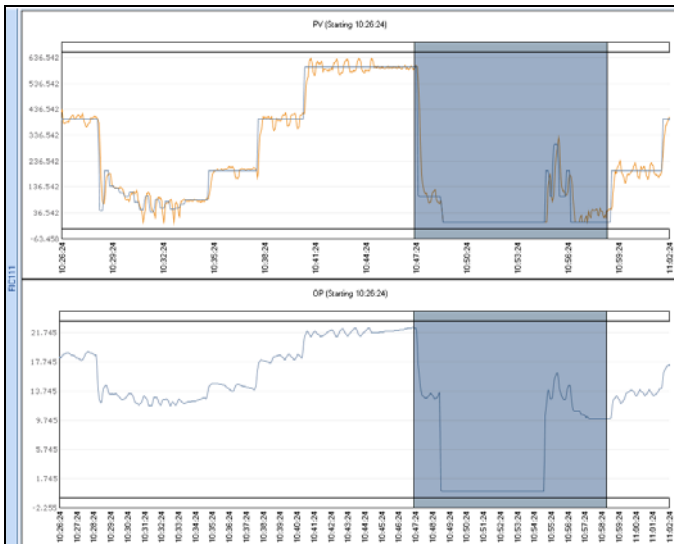
In the past, testing for non linearities (stiction, backlash, non linear gain, varying dead time ...) took time and required a series of bump tests prior to tuning the loop.

The robustness plot is based on model, tuning parameters and errors in model (error bound). Hence, robustness will include all defects. If error bound is high, then the user can analyze more in details only if stability margin has to be improved.

Even in presence of stiction for example, the model will be correctly identified.

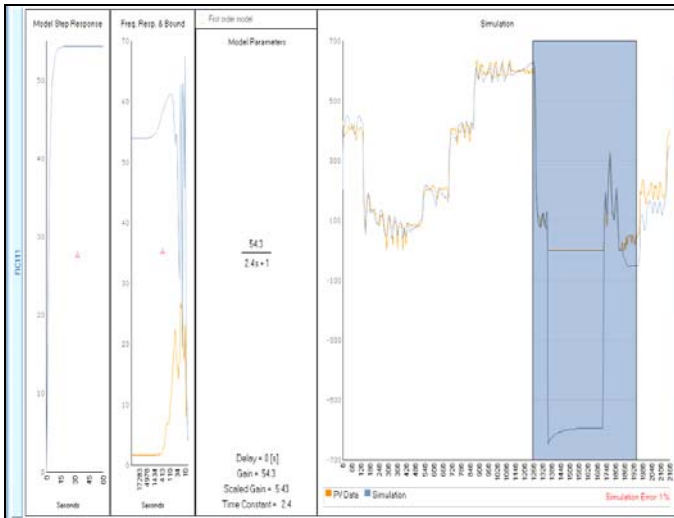
### AN EXAMPLE

A flow loop where setpoint changes occur is used to demonstrate how to tune this loop without extra tests.



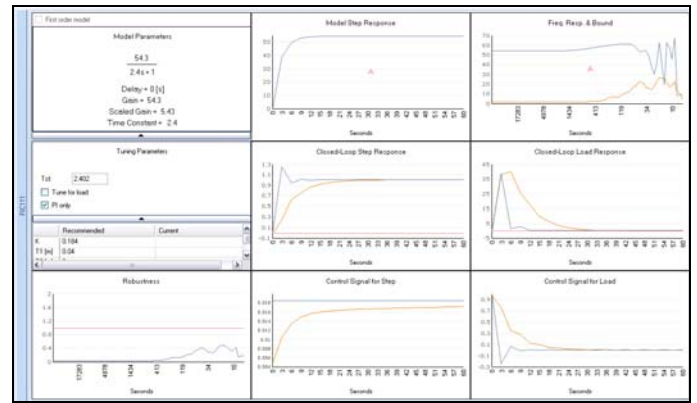
**Figure 9:** Data used for analyses. (The grey area has been sliced out since the pump was stopped.)

We observe very aggressive tuning on this loop. Most software will disregard this data and will be unable to modelize the process since oscillations and non linearities are present.



**Figure 10:** Model found, and uncertainty.

Strong non linearities are present but the software identifies a grade A model (high quality) with a small error.



**Figure 11:** Tuning pane and simulation.

Even with non linearities, those tuning parameters will maintain this loop stable in all situations.

## SUMMARY

Closed loop tuning has gained popularity since new tools are available. With skills and experience, it is possible to use these tools to tune loops and optimize processes without interfering with production. The process control engineer can set up and configure tests in advance and let the program generate generalized binary noise with appropriate amplitude and frequency. Many loops can be analyzed simultaneously.

These tests can be done during normal production, during night shifts without process control personnel or intervention by operations. Operations can stop the tests at anytime if they are uncomfortable with the program manipulating the setpoint slightly around its value.

It becomes realistic to tune all loops in a plant since a campaign can be launched at any time to analyze a group of loops.

Cascade loops, ratio control loops, and supervisory control loops are also tuned while in closed loop.

If the plant uses control performance monitoring tools, this software can identify which loops would benefit from retuning, and whether enough setpoint changes are already present to modelize and retune the loop using historical data.

All graphics have been generated using the Matrikon TaiJi-PID tuning tool.

## REFERENCES

Brittain, H. and Ruel, M. Optimize Your Process Using Normal Operation Data, NPRA Conference, Houston TX, October 2008

Brittain H., Dewey D., Ruel M. Closed Loop Tuning: Methodology, Tools, Benefits and Common Mistakes, ISA 2009 Conference, Houston Tx, October 2009

Ruel M., Control Valve Performance, Chemical Engineering, October 2000, pp 64-67

Zhu, Y.C. Multivariable process identification for MPC: the asymptotic method and its applications. Journal of Process Control (1998)., Vol. 8, No. 2, pp. 101-115.

Instrument Engineers' Handbook , chapter 5.6 Plantwide Control Loop Optimization (3rd Edition, edited by Bela G Liptak, CRC Press, Boca Raton FL, 2002), 40pp